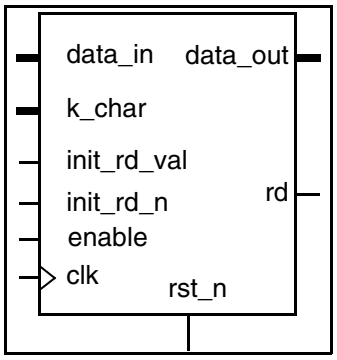
DW_8b10b_enc

8b10b Encoder

Version, STAR, and myDesignWare Subscriptions: IP Directory

Features and Benefits

- Configurable data width
- Configurable simplified Special Character control (for protocols requiring only the K28.5 special character)
- Synchronous initialization of Running Disparity with design specified value
- All outputs registered
- Provides minPower benefits (see Table 1-3 on page 3)



Revision History

Description

DW_8b10b_enc encodes 1 to 16 bytes of data using the 8b10b Direct Current (DC) balanced encoding scheme.

Table 1-1 Pin Description

Pin Name	Width	Direction	Function
clk	1 bit	Input	Clock
rst_n	1 bit	Input	Asynchronous reset, active low
init_rd_n	1 bit	Input	Synchronous initialization, active low
init_rd_val	1 bit	Input	Value of initial Running Disparity
k_char	*bytes* bits	Input	Special character controls (one control per byte to encode)
data_in	*bytes* × 8 bits	Input	Input data for encoding
rd	1 bit	Output	Current Running Disparity (before encoding data presented at `data_in`)
data_out	*bytes* × 10 bits	Output	8b10b encoded data
enable	1 bit	Input	Enables register clocking

Table 1-2 Parameter Description

Parameter	Value	Description
bytes	1 to 16 Default: 2	Number of bytes to encode
k28_5_only	0 or 1 Default: 0	Special character subset control parameter ■ 0 for all special characters available ■ 1 for only K28.5 available (when `k_char` = low, regardless of the value on `data_in`)
en_mode	0 or 1 Default: 0	Enable control ■ 0: The enable input port is not connected (backward compatible with older components) ■ 1: When `enable` = 0 the encoder is stalled
init_mode	0 or 1 Default: 0	Initialization mode for running disparity ■ 0: During active `init_rd_n` input, delay `init_rd_val` one clock cycle before applying it to `data_in` input in calculating `data_out` (backward-compatible with older components) ■ 1: During active `init_rd_n` input, directly apply `init_rd_val` to `data_in` input (with no clock cycle delay) in calculating data_out
rst_mode	0 or 1 Default: 0	Reset mode ■ 0: Asynchronous reset ■ 1: Synchronous reset
op_iso_mode	0 to 4 Default: 0	Operand isolation mode (controls datapath gating for minPower flow) Allows you to set the style of minPower datapath gating for this module ■ 0: Use the DW_lp_op_iso_mode[a] synthesis setting ■ 1: 'none' ■ 2: 'and' ■ 3: 'or' ■ 4: Preferred gating style: ‘and’ For details about enabling minPower datapath gating for this component, see “Enabling minPower” on page 17.

a. The DW_lp_op_iso_mode synthesis variable is available only in Design Compiler.
DW_lp_op_iso_mode sets a global style of datapath gating. To use the global style, set *op_iso_mode* to '0', Note that If the *op_iso_mode* parameter is set to '0' and DW_lp_op_iso_mode is either not set or set to 0', then no datapath gating is inserted for this component.

Table 1-3 Synthesis Implementations

Implementation Name	Function	License Feature Required
rtl	Synthesis model	DesignWare
lpwr[a]	Low Power synthesis model	■ DesignWare (P-2019.03 and later) ■ DesignWare-LP (before P-2019.03)

a. Requires that you enable minPower; for details, see "Enabling minPower" on page 17.
When minPower is enabled, the lpwr implementation is always chosen during synthesis.

Functional Operation

All outputs of the DW_8b10b_enc are registered and as such the encoded results of input data (on `data_in` and `k_char` ports) are not seen at the output ports `data_out` and `rd`, until after the rising edge of the `clk` input. The `init_rd_n` input synchronously initializes the running disparity to the value present on the `init_rd_val` port.

The code words generated are either perfectly balanced (i.e., they contain the same number of ones as zeros: 50% bias) or are unbalanced at 60% or 40% bias. A data byte that does not generate a balanced code generates one of two possible unbalanced code words (60% and 40%). The aggregate DC value is kept balanced by remembering the direction in which the last unbalanced code word was biased and generating the next unbalanced code word in the opposite bias direction. The one bit memory that recalls the bias of the last unbalanced code word is called the Running Disparity. A Running Disparity value of zero is synonymous with negative Running Disparity (–). A Running Disparity value of one is synonymous with positive Running Disparity (+).

Input data presented on the `data_in` port is encoded from the most significant 8-bit byte down to the least significant 8-bit byte. The most significant 8-bit byte on `data_in` is encoded to the most significant 10-bit byte on `data_out`. The second most significant 8-bit byte on `data_in` is encoded to the second most significant 10-bit byte on `data_out`. This continues down until finally the least significant 8-bit byte on `data_in` is encoded to the least significant 10-bit byte on `data_out`. The first bit of serial data to be transmitted from an encoded word is the most significant bit of `data_out` and the last serial bit to be transmitted is bit zero of `data_out`.

Reset

The `rst_n` input port asynchronously resets the Running Disparity (reflected on the `rd` output port) to zero (–) and the `data_out` port to all zeros, which is an invalid 8b10b code word.

Enable

The synthesis parameter *en_mode* determines the function of the input port `enable`. When *en_mode* = 1, DW_8b10b_enc uses the `enable` port to control clocking of registers in the module. With *en_mode* = 1, the module encodes information from `data_in` to `data_out` on every clock cycle when `enable` = 1. No encoding occurs for any clock cycles with `enable` = 0 (and *en_mode* = 1). When *en_mode* = 0 (which is the default used when *en_mode* is not set in the design), the `enable` input is not connected.

Running Disparity Initialization

The `init_rd_n` input is used to synchronously initialize the internal running disparity (reflected on the output port `rd`) to the value present on the port, `init_rd_val`. Initialization of running disparity does not cause the data output registers to be initialized. One interesting thing that `init_rd_n` and `init_rd_val` can be used for is to "disturb" the running disparity in order to inject errors into the encoded data stream. To inject a single event error, the application must effectively predict the next state of running disparity and initialize it to the opposite state. Such a prediction can be made either by prior knowledge of the data being sent (balanced data being sent implies the running disparity should not change) or to use the DW_8b10b_unbal component to predict whether the current state of running disparity will be flipped. Another less precise method of error injection would be to continuously initialize the running disparity while encoding. This causes all data words to be encoded with a constant initial running disparity.

The initialization of the running disparity behaves differently based on the synthesis parameter *init_mode*. When *init_mode* = 0 (the default and the backward compatible setting) and with `init_rd_n` asserted, the `init_rd_val` input is delayed one clock before it is applied to the `data_in` input for the encoded `data_out` output. Therefore, two clock cycles after asserting `init_rd_val`, the resulting `data_out`, of which `init_rd_val` is applied, becomes available. The `rd` output reflects the value of `init_rd_val` (with `init_rd_n` asserted) with a one-cycle latency (see Figure 1-1 on page 16). When *init_mode* = 1 and with `init_rd_n` asserted, the `init_rd_val` input is applied without any clock cycle delay to the `data_in` input. Thus, `data_out` reflected by `init_rd_val` being applied to `data_in` is available on the next clock cycle along with the newly calculated `rd` output (see Figure 1-2 on page 16). Note, any time `init_rd_n` is asserted when *init_mode* = 1, the `init_rd_val` input is applied to `data_in` for encoding, as opposed to when *init_mode* = 0, in which case the `rd` output is internally applied to `data_in`.

Special Character Encoding

Each byte of the port `data_in` has an associated special character control input on the input bus, `k_char`. When `k_char[0]` is low, data is encoded directly from `data_in`, bits 7 through 0, with no special characters generated. When `k_char[0]` is high, special characters are generated as per Table 1-4.

When the parameter *k28_5_only* is 0 and `k_char[`*i*`]` is high, characters encoded by synthetic model with any value other than the twelve listed in Table 1-4 are not guaranteed to be valid 8b10b code words. When simulation models simulate under this condition, they emit warning messages and the `data_out` port is set to Xs, as is the `rd` port.

Table 1-4 Special Character Encoding

Character Name	data_in value	k28_5_mode	Characteristics
K28.0	00011100 (1Ch)	0	Balanced
K28.1	00111100 (3Ch)	0	Unbalanced, comma
K28.2	01011100 (5Ch)	0	Unbalanced
K28.3	01111100 (7Ch)	0	Unbalanced
K28.4	10011100 (9Ch)	0	Balanced
K28.5	10111100 (BCh)	0	Unbalanced, comma

Table 1-4 Special Character Encoding (Continued)

Character Name	data_in value	k28_5_mode	Characteristics
K28.5	DON'T CARE	1	Unbalanced, comma
K28.6	11011100 (DCh)	0	Unbalanced
K28.7	11111000 (FCh)	0	Balanced, comma
K23.7	11110111 (F7h)	0	Balanced
K27.7	11111011 (FBh)	0	Balanced
K29.7	11111101 (FDh)	0	Balanced
K30.7	11111110 (FEh)	0	Balanced

Character Encoding

Table 1-5 shows the valid data character encoding for 8b10b.

Table 1-5 Valid Data Characters

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D0.0	000 00000	100111 0100	5	011000 1011	5	10	same
D1.0	000 00001	011101 0100	6	100010 1011	6	12	same
D2.0	000 00010	101101 0100	7	010010 1011	7	14	same
D3.0	000 00011	110001 1011	4	110001 0100	5	9	flip
D4.0	000 00100	110101 0100	7	001010 1011	7	14	same
D5.0	000 00101	101001 1011	6	101001 0100	7	13	flip
D6.0	000 00110	011001 1011	5	011001 0100	6	11	flip
D7.0	000 00111	111000 1011	4	000111 0100	4	8	flip
D8.0	000 01000	111001 0100	5	000110 1011	5	10	same
D9.0	000 01001	100101 1011	6	100101 0100	7	13	flip
D10.0	000 01010	010101 1011	7	010101 0100	8	15	flip
D11.0	000 01011	110100 1011	6	110100 0100	5	11	flip
D12.0	000 01100	001101 1011	5	001101 0100	6	11	flip
D13.0	000 01101	101100 1011	6	101100 0100	5	11	flip
D14.0	000 01110	011100 1011	5	011100 0100	4	9	flip

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D15.0	000 01111	010111 0100	6	101000 1011	6	12	same
D16.0	000 10000	011011 0100	6	100100 1011	6	12	same
D17.0	000 10001	100011 1011	4	100011 0100	5	9	flip
D18.0	000 10010	010011 1011	5	010011 0100	6	11	flip
D19.0	000 10011	110010 1011	6	110010 0100	5	11	flip
D20.0	000 10100	001011 1011	5	001011 0100	6	11	flip
D21.0	000 10101	101010 1011	8	101010 0100	7	15	flip
D22.0	000 10110	011010 1011	7	011010 0100	6	13	flip
D23.0	000 10111	111010 0100	5	000101 1011	5	10	same
D24.0	000 11000	110011 0100	5	001100 1011	5	10	same
D25.0	000 11001	100110 1011	6	100110 0100	5	11	flip
D26.0	000 11010	010110 1011	7	010110 0100	6	13	flip
D27.0	000 11011	110110 0100	5	001001 1011	5	10	same
D28.0	000 11100	001110 1011	5	001110 0100	4	9	flip
D29.0	000 11101	101110 0100	5	010001 1011	5	10	same
D30.0	000 11110	011110 0100	4	100001 1011	4	8	same
D31.0	000 11111	101011 0100	7	010100 1011	7	14	same
D0.1	001 00000	100111 1001	4	011000 1001	5	9	flip
D1.1	001 00001	011101 1001	5	100010 1001	6	11	flip
D2.1	001 00010	101101 1001	6	010010 1001	7	13	flip
D3.1	001 00011	110001 1001	4	110001 1001	4	8	same
D4.1	001 00100	110101 1001	6	001010 1001	7	13	flip
D5.1	001 00101	101001 1001	6	101001 1001	6	12	same
D6.1	001 00110	011001 1001	5	011001 1001	5	10	same
D7.1	001 00111	111000 1001	4	000111 1001	3	7	same
D8.1	001 01000	111001 1001	4	000110 1001	5	9	flip
D9.1	001 01001	100101 1001	6	100101 1001	6	12	same

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D10.1	001 01010	010101 1001	7	010101 1001	7	14	same
D11.1	001 01011	110100 1001	6	110100 1001	6	12	same
D12.1	001 01100	001101 1001	5	001101 1001	5	10	same
D13.1	001 01101	101100 1001	5	101100 1001	5	10	same
D14.1	001 01110	011100 1001	5	011100 1001	5	10	same
D15.1	001 01111	010111 1001	5	101000 1001	6	11	flip
D16.1	001 10000	011011 1001	5	100100 1001	6	11	flip
D17.1	001 10001	100011 1001	4	100011 1001	4	8	same
D18.1	001 10010	010011 1001	5	010011 1001	5	10	same
D19.1	001 10011	110010 1001	6	110010 1001	6	12	same
D20.1	001 10100	001011 1001	5	001011 1001	5	10	same
D21.1	001 10101	101010 1001	8	101010 1001	8	16	same
D22.1	001 10110	011010 1001	7	011010 1001	7	14	same
D23.1	001 10111	111010 1001	6	000101 1001	5	11	flip
D24.1	001 11000	110011 1001	4	001100 1001	5	9	flip
D25.1	001 11001	100110 1001	6	100110 1001	6	12	same
D26.1	001 11010	010110 1001	7	010110 1001	7	14	same
D27.1	001 11011	110110 1001	6	001001 1001	5	11	flip
D28.1	001 11100	001110 1001	5	001110 1001	5	10	same
D29.1	001 11101	101110 1001	6	010001 1001	5	11	flip
D30.1	001 11110	011110 1001	5	100001 1001	4	9	flip
D31.1	001 11111	101011 1001	6	010100 1001	7	13	flip
D0.2	010 00000	100111 0101	6	011000 0101	5	11	flip
D1.2	010 00001	011101 0101	7	100010 0101	6	13	flip
D2.2	010 00010	101101 0101	8	010010 0101	7	15	flip
D3.2	010 00011	110001 0101	6	110001 0101	6	12	same
D4.2	010 00100	110101 0101	8	001010 0101	7	15	flip

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D5.2	010 00101	101001 0101	8	101001 0101	8	16	same
D6.2	010 00110	011001 0101	7	011001 0101	7	14	same
D7.2	010 00111	111000 0101	4	000111 0101	5	9	same
D8.2	010 01000	111001 0101	6	000110 0101	5	11	flip
D9.2	010 01001	100101 0101	8	100101 0101	8	16	same
D10.2	010 01010	010101 0101	9	010101 0101	9	18	same
D11.2	010 01011	110100 0101	6	110100 0101	6	12	same
D12.2	010 01100	001101 0101	7	001101 0101	7	14	same
D13.2	010 01101	101100 0101	6	101100 0101	6	12	same
D14.2	010 01110	011100 0101	5	011100 0101	5	10	same
D15.2	010 01111	010111 0101	7	101000 0101	6	13	flip
D16.2	010 10000	011011 0101	7	100100 0101	6	13	flip
D17.2	010 10001	100011 0101	6	100011 0101	6	12	same
D18.2	010 10010	010011 0101	7	010011 0101	7	14	same
D19.2	010 10011	110010 0101	6	110010 0101	6	12	same
D20.2	010 10100	001011 0101	7	001011 0101	7	14	same
D21.2	010 10101	101010 0101	8	101010 0101	8	16	same
D22.2	010 10110	011010 0101	7	011010 0101	7	14	same
D23.2	010 10111	111010 0101	6	000101 0101	7	13	flip
D24.2	010 11000	110011 0101	6	001100 0101	5	11	flip
D25.2	010 11001	100110 0101	6	100110 0101	6	12	same
D26.2	010 11010	010110 0101	7	010110 0101	7	14	same
D27.2	010 11011	110110 0101	6	001001 0101	7	13	flip
D28.2	010 11100	001110 0101	5	001110 0101	5	10	same
D29.2	010 11101	101110 0101	6	010001 0101	7	13	flip
D30.2	010 11110	011110 0101	5	100001 0101	6	11	flip
D31.2	010 11111	101011 0101	8	010100 0101	7	15	flip

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D0.3	011 00000	100111 0011	4	011000 1100	4	8	flip
D1.3	011 00001	011101 0011	5	100010 1100	5	10	flip
D2.3	011 00010	101101 0011	6	010010 1100	6	12	flip
D3.3	011 00011	110001 1100	3	110001 0011	4	7	same
D4.3	011 00100	110101 0011	6	001010 1100	6	12	flip
D5.3	011 00101	101001 1100	5	101001 0011	6	11	same
D6.3	011 00110	011001 1100	4	011001 0011	5	9	same
D7.3	011 00111	111000 1100	3	000111 0011	3	6	same
D8.3	011 01000	111001 0011	4	000110 1100	4	8	flip
D9.3	011 01001	100101 1100	5	100101 0011	6	11	same
D10.3	011 01010	010101 1100	6	010101 0011	7	13	same
D11.3	011 01011	110100 1100	5	110100 0011	4	9	same
D12.3	011 01100	001101 1100	4	001101 0011	5	9	same
D13.3	011 01101	101100 1100	5	101100 0011	4	9	same
D14.3	011 01110	011100 1100	4	011100 0011	3	7	same
D15.3	011 01111	010111 0011	5	101000 1100	5	10	flip
D16.3	011 10000	011011 0011	5	100100 1100	5	10	flip
D17.3	011 10001	100011 1100	3	100011 0011	4	7	same
D18.3	011 10010	010011 1100	4	010011 0011	5	9	same
D19.3	011 10011	110010 1100	5	110010 0011	4	9	same
D20.3	011 10100	001011 1100	4	001011 0011	5	9	same
D21.3	011 10101	101010 1100	7	101010 0011	6	13	same
D22.3	011 10110	011010 1100	6	011010 0011	5	11	same
D23.3	011 10111	111010 0011	4	000101 1100	4	8	flip
D24.3	011 11000	110011 0011	4	001100 1100	4	8	flip
D25.3	011 11001	100110 1100	5	100110 0011	4	9	same
D26.3	011 11010	010110 1100	6	010110 0011	5	11	same

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D27.3	011 11011	110110 0011	4	001001 1100	4	8	flip
D28.3	011 11100	001110 1100	4	001110 0011	3	7	same
D29.3	011 11101	101110 0011	4	010001 1100	4	8	flip
D30.3	011 11110	011110 0011	3	100001 1100	3	6	flip
D31.3	011 11111	101011 0011	6	010100 1100	6	11	flip
D0.4	100 00000	100111 0010	5	011000 1101	5	10	same
D1.4	100 00001	011101 0010	6	100010 1101	6	12	same
D2.4	100 00010	101101 0010	7	010010 1101	7	14	same
D3.4	100 00011	110001 1101	4	110001 0010	5	9	flip
D4.4	100 00100	110101 0010	7	001010 1101	7	14	same
D5.4	100 00101	101001 1101	6	101001 0010	7	13	flip
D6.4	100 00110	011001 1101	5	011001 0010	6	11	flip
D7.4	100 00111	111000 1101	4	000111 0010	4	8	flip
D8.4	100 01000	111001 0010	5	000110 1101	5	10	same
D9.4	100 01001	100101 1101	6	100101 0010	7	13	flip
D10.4	100 01010	010101 1101	7	010101 0010	8	15	flip
D11.4	100 01011	110100 1101	6	110100 0010	5	11	flip
D12.4	100 01100	001101 1101	5	001101 0010	6	11	flip
D13.4	100 01101	101100 1101	6	101100 0010	5	11	flip
D14.4	100 01110	011100 1101	5	011100 0010	4	9	flip
D15.4	100 01111	010111 0010	6	101000 1101	6	12	same
D16.4	100 10000	011011 0010	6	100100 1101	6	12	same
D17.4	100 10001	100011 1101	4	100011 0010	5	9	flip
D18.4	100 10010	010011 1101	5	010011 0010	6	11	flip
D19.4	100 10011	110010 1101	6	110010 0010	5	11	flip
D20.4	100 10100	001011 1101	5	001011 0010	6	11	flip
D21.4	100 10101	101010 1101	8	101010 0010	7	15	flip

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D22.4	100 10110	011010 1101	7	011010 0010	6	13	flip
D23.4	100 10111	111010 0010	5	000101 1101	5	10	same
D24.4	100 11000	110011 0010	5	001100 1101	5	10	same
D25.4	100 11001	100110 1101	6	100110 0010	5	11	flip
D26.4	100 11010	010110 1101	7	010110 0010	6	13	flip
D27.4	100 11011	110110 0010	5	001001 1101	5	10	same
D28.4	100 11100	001110 1101	5	001110 0010	4	9	flip
D29.4	100 11101	101110 0010	5	010001 1101	5	10	same
D30.4	100 11110	011110 0010	4	100001 1101	4	8	same
D31.4	100 11111	101011 0010	7	010100 1101	7	14	same
D0.5	101 00000	100111 1010	5	011000 1010	6	11	flip
D1.5	101 00001	011101 1010	6	100010 1010	7	13	flip
D2.5	101 00010	101101 1010	7	010010 1010	8	15	flip
D3.5	101 00011	110001 1010	5	110001 1010	5	10	same
D4.5	101 00100	110101 1010	7	001010 1010	8	15	flip
D5.5	101 00101	101001 1010	7	101001 1010	7	14	same
D6.5	101 00110	011001 1010	6	011001 1010	6	12	same
D7.5	101 00111	111000 1010	5	000111 1010	4	9	same
D8.5	101 01000	111001 1010	5	000110 1010	6	11	flip
D9.5	101 01001	100101 1010	7	100101 1010	7	14	same
D10.5	101 01010	010101 1010	8	010101 1010	8	16	same
D11.5	101 01011	110100 1010	7	110100 1010	7	14	same
D12.5	101 01100	001101 1010	6	001101 1010	6	12	same
D13.5	101 01101	101100 1010	7	101100 1010	7	14	same
D14.5	101 01110	011100 1010	6	011100 1010	6	12	same
D15.5	101 01111	010111 1010	6	101000 1010	7	13	flip
D16.5	101 10000	011011 1010	6	100100 1010	7	13	flip

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D17.5	101 10001	100011 1010	5	100011 1010	5	10	same
D18.5	101 10010	010011 1010	6	010011 1010	6	12	same
D19.5	101 10011	110010 1010	7	110010 1010	7	14	same
D20.5	101 10100	001011 1010	6	001011 1010	6	12	same
D21.5	101 10101	101010 1010	9	101010 1010	9	18	same
D22.5	101 10110	011010 1010	8	011010 1010	8	16	same
D23.5	101 10111	111010 1010	7	000101 1010	6	13	flip
D24.5	101 11000	110011 1010	5	001100 1010	6	11	flip
D25.5	101 11001	100110 1010	7	100110 1010	7	14	same
D26.5	101 11010	010110 1010	8	010110 1010	8	16	same
D27.5	101 11011	110110 1010	7	001001 1010	6	13	flip
D28.5	101 11100	001110 1010	6	001110 1010	6	12	same
D29.5	101 11101	101110 1010	7	010001 1010	6	13	flip
D30.5	101 11110	011110 1010	6	100001 1010	5	11	flip
D31.5	101 11111	101011 1010	7	010100 1010	8	15	flip
D0.6	110 00000	100111 0110	5	011000 0110	4	9	flip
D1.6	110 00001	011101 0110	6	100010 0110	5	11	flip
D2.6	110 00010	101101 0110	7	010010 0110	6	13	flip
D3.6	110 00011	110001 0110	5	110001 0110	5	10	same
D4.6	110 00100	110101 0110	7	001010 0110	6	13	flip
D5.6	110 00101	101001 0110	7	101001 0110	7	14	same
D6.6	110 00110	011001 0110	6	011001 0110	6	12	same
D7.6	110 00111	111000 0110	3	000111 0110	4	7	same
D8.6	110 01000	111001 0110	5	000110 0110	4	9	flip
D9.6	110 01001	100101 0110	7	100101 0110	7	14	same
D10.6	110 01010	010101 0110	8	010101 0110	8	16	same
D11.6	110 01011	110100 0110	5	110100 0110	5	10	same

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D12.6	110 01100	001101 0110	6	001101 0110	6	12	same
D13.6	110 01101	101100 0110	5	101100 0110	5	10	same
D14.6	110 01110	011100 0110	4	011100 0110	4	8	same
D15.6	110 01111	010111 0110	6	101000 0110	5	11	flip
D16.6	110 10000	011011 0110	6	100100 0110	5	11	flip
D17.6	110 10001	100011 0110	5	100011 0110	5	10	same
D18.6	110 10010	010011 0110	6	010011 0110	6	12	same
D19.6	110 10011	110010 0110	5	110010 0110	5	10	same
D20.6	110 10100	001011 0110	6	001011 0110	6	12	same
D21.6	110 10101	101010 0110	7	101010 0110	7	14	same
D22.6	110 10110	011010 0110	6	011010 0110	6	12	same
D23.6	110 10111	111010 0110	5	000101 0110	6	11	flip
D24.6	110 11000	110011 0110	5	001100 0110	4	9	same
D25.6	110 11001	100110 0110	5	100110 0110	5	10	same
D26.6	110 11010	010110 0110	6	010110 0110	6	12	same
D27.6	110 11011	110110 0110	5	001001 0110	6	11	flip
D28.6	110 11100	001110 0110	4	001110 0110	4	8	same
D29.6	110 11101	101110 0110	5	010001 0110	5	10	flip
D30.6	110 11110	011110 0110	4	100001 0110	5	9	flip
D31.6	110 11111	101011 0110	7	010100 0110	6	13	flip
D0.7	111 00000	100111 0001	4	011000 1110	4	8	same
D1.7	111 00001	011101 0001	5	100010 1110	5	10	same
D2.7	111 00010	101101 0001	6	010010 1110	6	12	same
D3.7	111 00011	110001 1110	3	110001 0001	4	7	flip
D4.7	111 00100	110101 0001	5	001010 1110	6	11	same
D5.7	111 00101	101001 1110	5	101001 0001	6	11	flip
D6.7	111 00110	011001 1110	4	011001 0001	5	9	flip

Table 1-5 Valid Data Characters (Continued)

Data Byte Name	Bits HGF EDCBA	Current RD – abcdei fghj	RD – Transition Density	Current RD + abcdei fghj	RD + Transition Density	Combined Transition Density	Ending RD
D7.7	111 00111	111000 1110	3	000111 0001	3	6	flip
D8.7	111 01000	111001 0001	4	000110 1110	4	8	same
D9.7	111 01001	100101 1110	5	100101 0001	6	11	flip
D10.7	111 01010	010101 1110	6	010101 0001	7	13	flip
D11.7	111 01011	110100 1110	5	110100 1000	5	10	flip
D12.7	111 01100	001101 1110	4	001101 0001	5	9	flip
D13.7	111 01101	101100 1110	5	101100 1000	5	10	flip
D14.7	111 01110	011100 1110	4	011100 1000	4	8	flip
D15.7	111 01111	010111 0001	5	101000 1110	5	10	same
D16.7	111 10000	011011 0001	5	100100 1110	5	10	same
D17.7	111 10001	100011 0111	4	100011 0001	4	8	flip
D18.7	111 10010	010011 0111	5	010011 0001	5	10	flip
D19.7	111 10011	110010 1110	5	110010 0001	4	9	flip
D20.7	111 10100	001011 0111	5	001011 0001	5	10	flip
D21.7	111 10101	101010 1110	7	101010 0001	6	13	flip
D22.7	111 10110	011010 1110	6	011010 0001	5	11	flip
D23.7	111 10111	111010 0001	4	000101 1110	4	8	same
D24.7	111 11000	110011 0001	4	001100 1110	4	8	same
D25.7	111 11001	100110 1110	5	100110 0001	4	9	flip
D26.7	111 11010	010110 1110	6	010110 0001	5	11	flip
D27.7	111 11011	110110 0001	4	001001 1110	4	8	same
D28.7	111 11100	001110 1110	4	001110 0001	3	7	flip
D29.7	111 11101	101110 0001	4	010001 1110	4	8	same
D30.7	111 11110	011110 0001	3	100001 1110	3	6	same
D31.7	111 11111	101011 0001	6	010100 1110	6	12	same

Suppressing Warning Messages During Verilog Simulation

The Verilog simulation model includes macros that allow you to suppress warning messages during simulation.

To suppress all warning messages for all DWBB components, define the DW_SUPPRESS_WARN macro in either of the following ways:

- Specify the Verilog preprocessing macro in Verilog code:

  ```
  `define DW_SUPPRESS_WARN
  ```

- Or, include a command line option to the simulator, such as:

  `+define+DW_SUPPRESS_WARN` (which is used for the Synopsys VCS simulator)

The warning messages for this model include the following:

- If values other than 1 or 0 are present on a clock port, the following message is displayed:

  ```
  WARNING: <instance_path>.<clock_name>_monitor:
      at time = <timestamp>, Detected unknown value, x, on <clock_name> input.
  ```

  To suppress only this warning message for all DWBB components, use the following macro:

  - Define the DW_DISABLE_CLK_MONITOR macro. You can define this macro in the following ways:
    - Specify the Verilog preprocessing macro in Verilog code:

      ```
      `define DW_DISABLE_CLK_MONITOR
      ```

    - Or, include a command line option to the simulator, such as:

      `+define+DW_DISABLE_CLK_MONITOR` (which is used for the Synopsys VCS simulator)

  This message is also suppressed using the DW_SUPPRESS_WARN macro explained earlier.

- If `data_in` contains invalid data for a specific configuration, the following message is displayed:

  ```
  WARNING: <instance_path>:
      at time = <timestamp>, Data on DW_8b10b_enc's data_in is invalid for k_char=1 and
  k28_5_only=0.
  ```

  To suppress this message, use the DW_SUPPRESS_WARN macro explained earlier.

Timing Waveforms

Figure 1-1 Running Disparity Initialization When init_mode=0

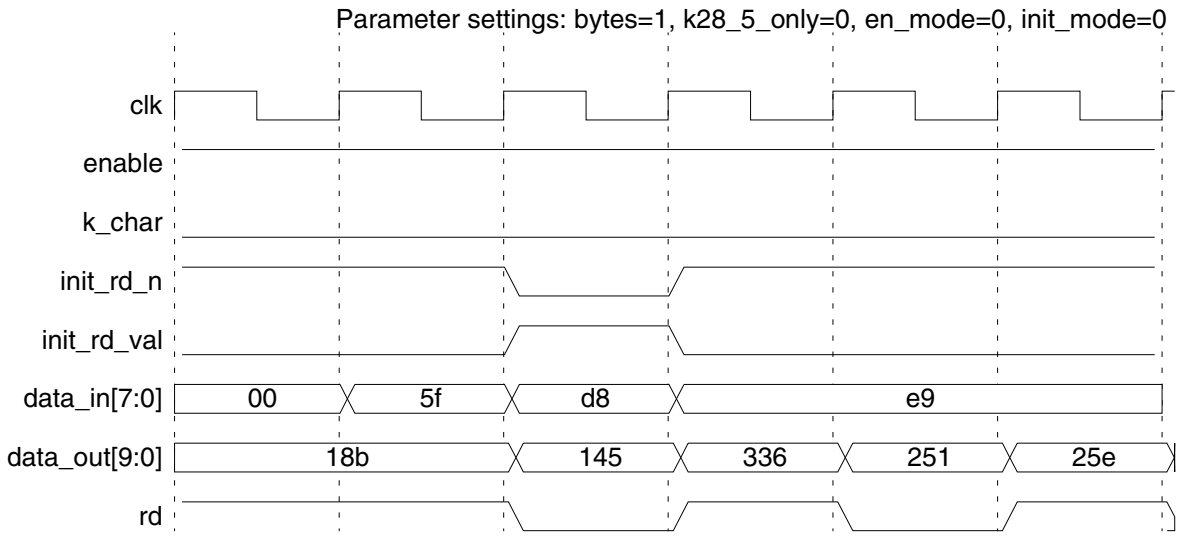
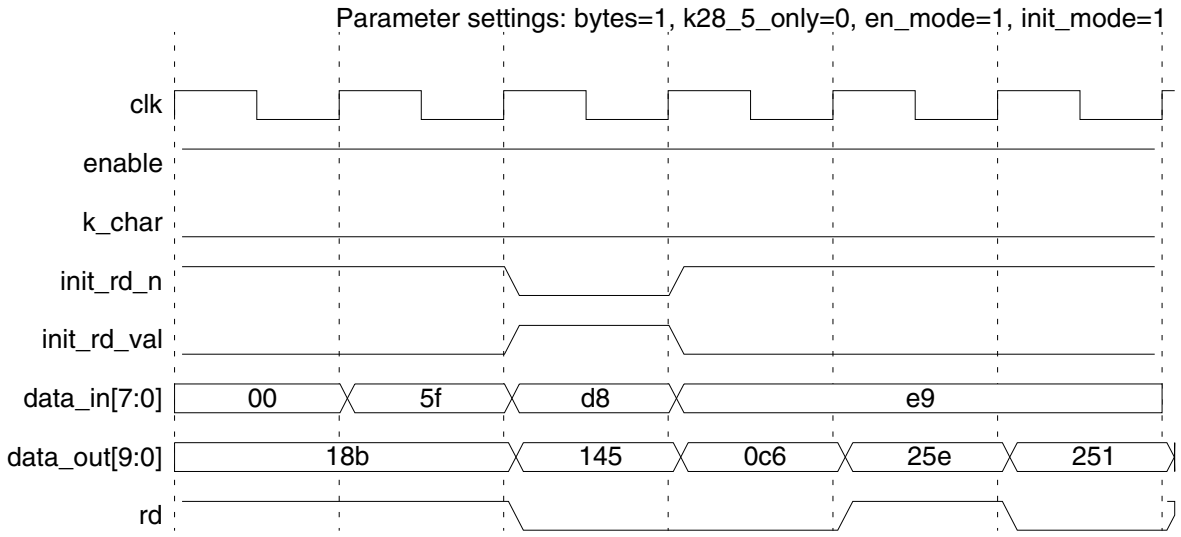


Figure 1-2 Running Disparity Initialization When init_mode=1



Enabling minPower

You can instantiate this component without enabling minPower, but to achieve datapath gating power savings, you must enable minPower optimization, as follows:

- Design Compiler
  - Version P-2019.03 and later:

    ```
    set power_enable_minpower true
    ```

  - Before version P-2019.03 (requires the DesignWare-LP license feature):

    ```
    set synthetic_library {dw_foundation.sldb dw_minpower.sldb}
    set link_library {* $target_library $synthetic_library}
    ```

- Fusion Compiler

  Optimization for minPower is enabled as part of the total_power metric setting. To enable the total_power metric, use the following:

  ```
  set_qor_strategy -stage synthesis -metric total_power
  ```

Related Topics

- Coding Group – Coding Overview
- DesignWare Building Block IP User Guide

HDL Usage Through Component Instantiation - VHDL

```
library IEEE,DWARE;
use IEEE.std_logic_1164.all;
use DWARE.DWpackages.all;
use DWARE.DW_foundation_comp.all;

entity DW_8b10b_enc_inst is
  generic (inst_bytes       : integer := 2;
           inst_k28_5_only  : integer := 0;
           inst_en_mode     : integer := 1;
           inst_init_mode   : integer := 1;
           inst_rst_mode    : integer := 0;
           inst_op_iso_mode : integer := 0 );
  port (inst_clk          : in std_logic;
        inst_rst_n        : in std_logic;
        inst_init_rd_n    : in std_logic;
        inst_init_rd_val  : in std_logic;
        inst_k_char       : in std_logic_vector(inst_bytes-1 downto 0);
        inst_data_in      : in std_logic_vector(inst_bytes*8-1 downto 0);
        rd_inst           : out std_logic;
        data_out_inst     : out std_logic_vector(inst_bytes*10-1 downto 0);
        inst_enable       : in std_logic );

end DW_8b10b_enc_inst;

architecture inst of DW_8b10b_enc_inst is
begin
  -- Instance of DW_8b10b_enc
  U1 : DW_8b10b_enc
    generic map (bytes => inst_bytes,   k28_5_only => inst_k28_5_only,
                 en_mode => inst_en_mode, init_mode => inst_init_mode,
                 rst_mode => inst_rst_mode, op_iso_mode => inst_op_iso_mode )
    port map (clk => inst_clk,   rst_n => inst_rst_n,
              init_rd_n => inst_init_rd_n,   init_rd_val => inst_init_rd_val,
               k_char => inst_k_char,   data_in => inst_data_in,
               rd => rd_inst,   data_out => data_out_inst,
               enable => inst_enable );
end inst;

-- Configuration for use with VHDL simulator
-- pragma translate_off
configuration DW_8b10b_enc_inst_cfg_inst of DW_8b10b_enc_inst is
  for inst
  end for; -- inst
end DW_8b10b_enc_inst_cfg_inst;
-- pragma translate_on
```

HDL Usage Through Component Instantiation - Verilog

```
module DW_8b10b_enc_inst(inst_clk, inst_rst_n, inst_init_rd_n,
                         inst_init_rd_val, inst_k_char,
                         inst_data_in, rd_inst, data_out_inst, inst_enable );
  parameter inst_bytes = 2;
  parameter inst_k28_5_only = 0;
  parameter inst_en_mode = 1;
  parameter inst_init_mode = 1;
  parameter inst_rst_mode = 0;
  parameter inst_op_iso_mode = 0;

  input inst_clk;
  input inst_rst_n;
  input inst_init_rd_n;
  input inst_init_rd_val;
  input [inst_bytes-1 : 0] inst_k_char;
  input [inst_bytes*8-1 : 0] inst_data_in;
  output rd_inst;
  output [inst_bytes*10-1 : 0] data_out_inst;
  input inst_enable;

  // Instance of DW_8b10b_enc
  DW_8b10b_enc #(inst_bytes, inst_k28_5_only, inst_en_mode,
                 inst_init_mode, inst_rst_mode, inst_op_iso_mode)
    U1 (.clk(inst_clk),   .rst_n(inst_rst_n),   .init_rd_n(inst_init_rd_n),
        .init_rd_val(inst_init_rd_val),   .k_char(inst_k_char),
        .data_in(inst_data_in),   .rd(rd_inst),   .data_out(data_out_inst),
        .enable(inst_enable)  );
endmodule
```

Revision History

For notes about this release, see the *DesignWare Building Block IP Release Notes*.

For lists of both known and fixed issues for this component, refer to the STAR report.

For a version of this datasheet with visible change bars, click here.

Date	Release	Updates
July 2020	DWBB_201912.5	■ Adjusted content and title of "Suppressing Warning Messages During Verilog Simulation" on page 15 and added the DW_SUPPRESS_WARN macro
October 2019	DWBB_201903.5	■ Added the "Disabling Clock Monitor Messages" section
March 2019	DWBB_201903.0	■ Clarified the op_iso_mode parameter in Table 1-2 on page 2 ■ Clarified license requirements in Table 1-3 on page 3 ■ Added "Enabling minPower" on page 17
January 2019	DWBB_201806.5	■ Updated example in "HDL Usage Through Component Instantiation - VHDL" on page 18 ■ Added this Revision History table and the document links on this page

Copyright Notice and Proprietary Information

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at https://www.synopsys.com/company/legal/trademarks-brands.html.

All other product or company names may be trademarks of their respective owners.

Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
www.synopsys.com